Securing Credential Management within Proxied Digital Wallet

Nigel Sahl[1], Louis Caesa Kesuma[2], Addin Munawwar Yusuf[3],

Antonio Natthan Krishna[4], Jong Uk Choi[5]

MarkAny Inc.

{nigel[1], louis.c.k[2], addin[3], neo[4]}@ganeshait.com, juchoi[5]@markany.com

Abstract

While decentralized identity architectures offer enhanced privacy through Self-Sovereign Identity (SSI), practical mobile implementations often require intermediary servers to ensure availability. However, introducing a proxy server creates significant security concerns regarding data confidentiality and trust. This paper presents a comprehensive security framework for a proxied digital wallet designed to maintain the cryptographic integrity of Verifiable Credentials (VC). We detail a dual-key infrastructure that strictly separates Signing Keys from Encryption Keys, ensuring the proxy facilitates transport without accessing plaintext data. By enforcing End-to-End Encryption (E2EE) and ephemeral tokenization, the architecture mitigates replay attacks while supporting privacy-preserving mechanisms like Zero-Knowledge Proofs. Our analysis demonstrates that a proxied architecture can satisfy Web 3.0 security requirements without compromising user sovereignty.

Ⅰ. Introduction

Advances in decentralized architectures utilizing Decentralized Identifiers (DIDs) [1] and Verifiable Credentials (VCs) [2] offer a path toward tamper-proof identity management. These systems strengthen identity ownership by ensuring cryptographic integrity independent of central authorities.

However, fully decentralized, peer-to-peer (P2P) mobile wallets face significant hurdles in real-world deployment. Dynamic network conditions, NAT traversal issues, and battery constraints often impede reliable, persistent connections [3] Consequently, many systems rely on intermediary servers to guarantee message delivery. The introduction of a proxy server, however, reintroduces centralization risks, particularly regarding data confidentiality. If a proxy is compromised, user credentials could be exposed. Furthermore, vulnerabilities such as weak randomness in key generation on mobile devices can lead to impersonation, allowing malicious actors to replicate access keys [4].

This study proposes a security baseline for a [5]. We introduce a blind-routing architecture where the backend relay manages transport metadata but is cryptographically prevented from accessing credential payloads via strict End-to-End Encryption (E2EE) and a dual-key strategy.

Ⅱ. Proxied System Design

A. Architecture and Entities

The system adopts a Proxy Design Pattern (Fig. 1). A Proxy Server acts as a high-availability mediator between the three key W3C entities:

- **Holder:** Stores credentials and initiates proofs.
- **Issuer:** Cryptographically signs and issues VCs.
- **Verifier:** Requests and validates proofs.

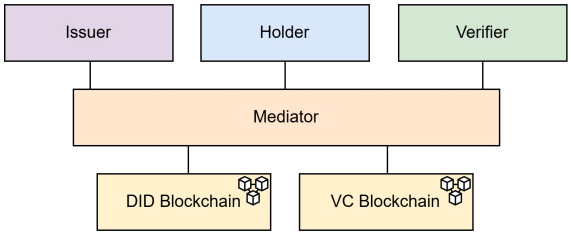

Fig. 1. Proxied System Architecture: The proxy routes encrypted packages without access to K_{Enc}^{Priv}.

This setup allows the system to balance the routing efficiency of a centralized server with the data ownership principles of SSI.

B. Cryptographic Artifacts

To isolate security domains, each entity maintains two distinct key pairs derived from a single BIP39 seed:

- **Signing Key** (K_{Sign}): Used solely for proving identity and signing Verifiable Presentations (VP).
- **Encryption Key** (K_{Enc}): Used solely for E2EE, ensuring confidentiality during transport.

The system utilizes DIDs to resolve these public keys via an immutable ledger, ensuring no central authority controls the root of trust.

Ⅲ. Securing the Wallet

A. Identity Generation

To establish a root of trust, we implement a Hierarchical Deterministic key strategy where a BIP39 mnemonic generates a Master Seed via PBKDF2-HMAC-SHA512. From this seed, K_{Sign} and K_{Enc} are derived using distinct BIP32 paths for

hierarchical recovery and cryptographic isolation. EdDSA (Ed25519) is used for signing because its deterministic nature avoids risks from weak random nonces in ECDSA and improves verification performance [5].

B. Credential Authenticity

Authenticity is managed via a dual-layer signing process. First, the Issuer signs the VC with $\text{Issuer}K^{Priv}_{Sign}$ to certify data integrity. Second, during presentation, the Holder signs the VP with $\text{Holder}K^{Priv}_{Sign}$. This ensures non-repudiation, preventing the Issuer from denying issuance, and the Holder from denying presentation.

C. Secure Sharing Workflow (E2EE)

To allow the proxy to route messages without inspecting them, we enforce the following E2EE workflow:

1. Issuance: The Holder requests a credential. The Issuer encrypts the signed VC using the Holder's Public Encryption Key $\text{Holder}K^{Pub}_{Enc}$. The proxy forwards this blob, which only the Holder can decrypt.
2. Presentation: The presentation flow separates content generation (Privacy) from transport (Security):
    a. Payload Generation: The Holder generates a VP. This can be a full disclosure, or a privacy-preserving Zero-Knowledge Proof (ZKP) (e.g., proving Age > 18 without revealing birthdate).
    b. Encryption: The Holder encrypts the signed VP using the Verifier's Public Key $\text{Verifier}K^{Pub}_{Enc}$.
    c. Tokenization: The encrypted payload is uploaded to the proxy, which returns a unique ephemeral VP ID.
    d. Retrieval: The Verifier scans a QR code containing the VP ID, fetches the encrypted blob, and decrypts it using $\text{Verifier}K^{Priv}_{Enc}$.
    e. Secure Deletion: Upon successful retrieval, the proxy immediately deletes the data, preventing replay attacks.

D. Revocation

Revocation is handled via Cryptographic Accumulators on the blockchain. The Holder maintains a "witness" value. During verification, the Verifier checks the witness against the on-chain accumulator. This proves credential validity without requiring the Verifier to download the full list of revoked IDs, preserving privacy [6].

IV. Evaluation

A. Security and Privacy Analysis

The proposed architecture addresses critical threats through three mechanisms. Sovereignty is enforced by local key generation in the device Secure Enclave to ensure private keys remain under user control. Blind routing is achieved through end-to-end encryption so the proxy only accesses transport metadata. Resilience is provided by key rotation protocols that allow users to update DID Documents without sacrificing identity.

B. Storage and Bandwidth

The use of Ed25519 provides compact 32-byte keys and 64-byte signatures, significantly reducing storage compared to RSA. Bandwidth efficiency is maintained by serializing credentials as JWTs which remain small enough for mobile transmission even after encryption [7].

V. Discussion

A. P2P vs. Proxied Architecture

Pure P2P models require a "coincidence of availability" where both parties must be online simultaneously, often failing due to NAT traversal and dynamic IPs. Our architecture adopts the Mediator pattern [8], queuing encrypted messages to achieve Web 2.0 reliability while maintaining a Web 3.0 security profile.

B. Post-Quantum Preparedness

Standard encryption (RSA, ECC) faces existential risks from Shor's algorithm [9]. To counter "Harvest Now, Decrypt Later" threats, our framework supports cryptographic agility, enabling the K_{Enc} pair to be upgraded to Post-Quantum Cryptography (PQC) standards, such as lattice-based algorithms, without altering the underlying DID infrastructure [10].

VI. Conclusion

This paper presented a framework for securing proxied digital wallets. We demonstrated that by strictly separating Identity Keys from Encryption Keys and enforcing E2EE, a proxy server can facilitate reliable credential exchange without compromising user privacy. The integration of HD keys, dual-layer signing, and support for ZKPs provides a robust foundation for non-repudiation. Future work will focus on implementing lattice-based PQC to secure the transport layer against quantum adversaries.

ACKNOWLEDGMENT

The authors thank the R&D team at MarkAny GaneshaIT Indonesia for their support. This research was made possible through the technical resources, professional guidance, and collaborative environment provided by the organization.

REFERENCES

[1] M. Sporny et al., "Decentralized identifiers (dids) v1.0," W3C, Recom-mendation, 2022, https://www.w3.org/TR/did-1.0/.

[2] B. Podgorelec, L. Alber, and T. Zefferer, “What is a (digital) identity wallet? a systematic literature review,” in 2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC), 2022, pp. 809–818.

[3] O. Kassinen, “Efficient middleware and resource management in mobile peer-to-peer systems,” Ph.D. dissertation, University of Oulu, 2011, faculty of Technology. [Online]. Available: https://urn.fi/URN:ISBN:9789514295737

[4] C. Shaik, “Unforgettable user defined seed phrase for cryptocurrency wallets,” International Journal on Cryptography and Information Security (IJCIS), vol. 10, no. 4, 2020.

[5] S. Josefsson and I. Liusvaara, “Edwards-curve digital signature algorithm (eddsa),” Internet Research Task Force (IRTF), Request for Comments 8032, Jan. 2017, informational. [Online]. Available: https://www.rfc-editor.org/rfc/rfc8032

[6] B. Kim, Y. Cho, S. Kim, H. Kim, and S. Woo, “A security analysis of blockchain-based did services,” IEEE Access, vol. 9, pp. 22 894–22 913, 2021.

[7] M. Jones, J. Bradley, and N. Sakimura, “JSON Web Token (JWT),”2015. [Online]. Available: https://www.rfc-editor.org/rfc/rfc7519.html

[8] P. Windley, “The sovrin ssi stack,”https://www.windley.com/archives/2020/03/the sovrin ssi stack.shtml, 2020, accessed: Dec. 5, 2025.

[9] L. Wang, X. Shen, J. Li, J. Shao, and Y. Yang, “Cryptographic primitives in blockchains,” Journal of Network and Computer Applications, vol. 127, pp. 43–58, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S10848 0451830362X

[10] S. Al-Janabi, “Post-quantum blockchain: Challenges and opportunities,”2025. [Online]. Available: https://arxiv.org/abs/2508.17071